

1 11. The method of claim 7 wherein the second set of low-level instructions is a test-
2 and-set instruction.

1 12. An apparatus comprising:
2 a memory including a shared memory location;
3 a translation unit coupled with the memory, the translation unit to translate a first
4 program unit including a memory update operation to be performed
5 atomically into a second program unit upon determining that a set of one
6 or more low-level instructions support a data size for the memory update
7 operation, the second program unit to associate the set of low-level
8 instructions with the memory update operation, the set of low-level
9 instructions to ensure atomicity of the memory update operation;
10 a compiler unit coupled with the translation unit and the shared-memory, the
11 compiler unit to compile the second program unit; and
12 a linker unit coupled with the compiler unit and the shared-memory, the linker
13 unit to link the compiled second program unit with a library.

1 13. The apparatus of claim 12 wherein the second program unit to associate the set of
2 low-level instructions with the memory update operation comprises encapsulating the
3 memory update operation.

1 14. The apparatus of claim 12 further comprising a set of one or more processors to
2 host a plurality of threads, the plurality of threads to execute the second program unit.

1 15. The apparatus of claim 12 wherein the second program unit to associate the set of
2 low-level instructions with the memory update operation comprises the translation unit to
3 generate a callback routine enclosing the memory update operation and the translation
4 unit to encapsulate the callback routine with a routine for the set of low-level instructions.

1 16. A system comprising:
2 a memory including a shared memory location;
3 a translation unit coupled with the shared-memory, the translation unit to translate
4 a first program unit including a memory update operation to be performed
5 atomically into a second program unit upon determining that a set of one
6 or more low-level instructions support a data size for the memory update
7 operation, the second program unit to associate the set of low-level
8 instructions with the memory update operation, the set of low-level
9 instructions to ensure atomicity of the memory update operation;
10 a compiler unit coupled with the translation unit and the shared-memory, the
11 compiler unit to compile the second program unit; and
12 a set of one or more processors coupled with the shared-memory, the translation
13 unit, and the compiler unit, the set of processors to host a plurality of
14 threads, the plurality of threads to perform the memory update operation in
15 accordance with the set of low-level instructions.

1 17. The system of claim 16 wherein the second program unit to associate the set of
2 low-level instructions with the memory update operation comprises encapsulating the
3 memory update operation.

1 18. The system of claim 16 wherein each of the set of processors comprise:
 2 a first register coupled with the shared-memory, the first register to host a first
 3 value loaded by one of the plurality of threads from the shared memory
 4 location; and
 5 a second register coupled with the shared-memory, the second register to host a
 6 result of generated by the one of the plurality of threads executing the
 7 memory update operation.

1 19. The system of claim 16 wherein the second program unit to associate the set of
 2 low-level instructions with the memory update operation comprises the translation unit to
 3 generate a callback routine enclosing the memory update operation and the translation
 4 unit to encapsulate the callback routine with a routine for the set of low-level instructions.

1 20. A machine-readable medium that provides instructions, which when executed by a
 2 set of one or more processors, cause said set of processors to perform operations
 3 comprising:
 4 receiving a first program unit in a parallel computing environment, the first
 5 program unit including a memory update operation to be performed
 6 atomically, the memory update operation having an operand, the operand
 7 being of a data-type and of a data size; and
 8 translating the first program unit into a second program unit, the second program
 9 unit to associate the memory update operation with a set of one or more
 10 low-level instructions upon determining that the data size of the operand is